

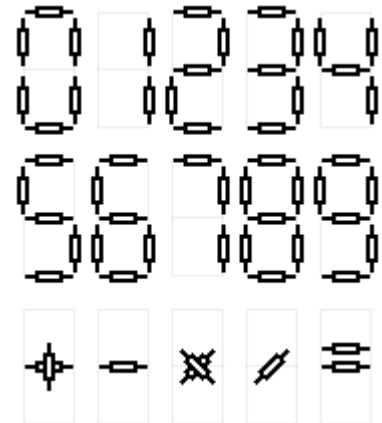
А. Тысяча резисторов

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Дождаясь преподавателя после занятий, Егор Дувайлов решил прогуляться по учебному корпусу X-OSPU (eXtreme Open Source Programming University). Зайдя в одну из открытых аудиторий, он увидел на партах кучу каких-то маленьких штук. «Наверное, радиодетали», — догадался Егор. Подойдя к одной из парт, он увидел приоткрытый спичечный коробок с такими же штуками и с надписью «Резисторы 1к». «Неужели в такую маленькую коробочку помещается целая тысяча этих... резисторов...», — удивился Егор.

Посмотрев на соседние парты, Егор увидел, что резисторы на них разложены далеко не случайным образом. Из резисторов были сложены символы — цифры и знаки математических операций (как показано на картинке), а из этих символов были составлены математические выражения вида $A \text{ op } B = C$. «Ага! Так вот, как делают калькуляторы!», — подумал Егор. Но тут он заметил, что одно из выражений — неправильное. «А этот калькулятор, похоже, получился бракованным. Надо его починить», — решил Егор.

В радиоэлектронике, как вы уже поняли, Егор не силён, поэтому, чтобы ничего не испортить, он решил пользоваться следующими правилами:



- для исправления выражения можно переложить ровно один резистор
- резистор можно вынуть из любого символа (цифры, знака операции или даже знака равенства)
- вынутый резистор нужно положить на любое свободное место в любом уже имеющемся символе (в том же символе, из которого его вынули, или в любом другом)
- количество символов в выражении должно остаться таким же, как было
- изменённые в результате перекалывания символы должны в результате выглядеть в точности так, как показано на картинке
- в получившемся выражении должен быть хотя бы один знак равенства
- если в получившемся выражении будет деление, то числа должны делиться нацело
- получившееся выражение должно иметь математический смысл и быть истинным

Помогите Егору исправить выражение или сообщите ему, что сделать это по его правилам невозможно. Если существует несколько способов исправить выражение, найдите любой.

Входные данные

В единственной строке находится выражение, которое необходимо исправить, в формате $A \text{ op } B = C$. Числа A , B и C являются целыми, неотрицательными, могут содержать от 1 до 9 цифр и могут содержать ведущие нули. Знак операции (op) может быть «+», «-», «x» (маленькая английская икс; умножить) или «/» (разделить). Пробелы между числами и знаками отсутствуют. Гарантируется, что формат выражения корректен, но само выражение — неправильное.

Выходные данные

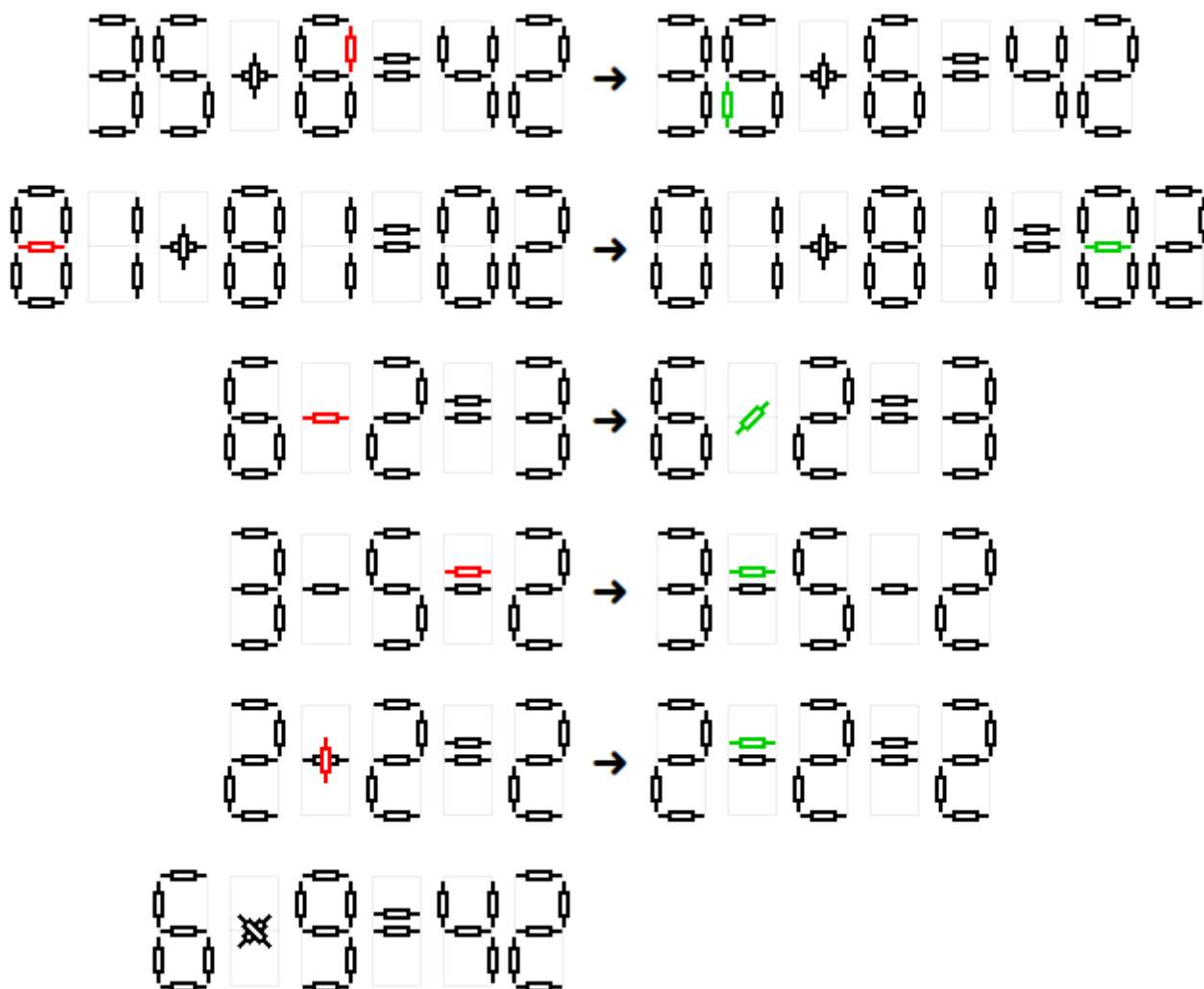
Если исправить выражение по правилам Егора возможно, выведите исправленное выражение в формате $A' \text{ op}_1 B' \text{ op}_2 C'$, где A' , B' и C' — целые числа, а op_1 и op_2 — знаки «+», «-», «x», «/» или «=».

Если исправить выражение невозможно, выведите «Impossible».

Пример

| Входные данные | Результат |
|-----------------|------------|
| $35+8=42$ | $36+6=42$ |
| $81+81=02$ | $01+81=82$ |
| $6-2=3$ | $6/2=3$ |
| $3-5=2$ | $3=5-2$ |
| $2+2=2$ | $2=2=2$ |
| $6 \times 9=42$ | Impossible |

Ниже приведены картинки, соответствующие выражениям из примеров. Красным и зелёным показаны резисторы, которые переложили.





В. Законопроект

Ограничение на время работы: для C и C++ — 1 секунда, для Java и Kotlin — 4 секунды
Ограничение на использование памяти: 256МБ

Олигарху Василию для расширения своего бизнеса необходимы изменения в законе. Для этого он решил подкупить некоторых депутатов, чтобы те проголосовали за нужный законопроект. Чтобы убрать сомнения у населения в важности данного закона, он решил получить 100% голосов.

Верховный совет состоит из N депутатов, которые представляют собой иерархическую структуру. Это означает, что есть ровно один депутат, который имеет влияние на всех остальных. У него в подчинении есть некоторое количество депутатов, которые непосредственно отчитываются ему. У каждого из них тоже есть подчиненные (но не обязательно). Как можно догадаться — депутаты являют собой древовидную структуру, где в корне — самый влиятельный депутат. Депутат A влияет на депутата B , если A находится в дереве выше и является предком B (не обязательно прямым).

Чтобы проголосовать за законопроект, для депутата i должно выполняться одно из условий:

- получить оплату A_i от Василия и тайно проголосовать
- получить оплату B_i от Василия и публично выступить за поддержку законопроекта (и проголосовать соответственно)
- получить оплату C_i и проголосовать под влиянием — для этого необходимо найти хотя-бы одного депутата, который влияет на него (является его предком в дереве) и публично выступит за поддержку законопроекта

Василий хочет подкупить депутатов так, чтобы минимизировать затраты и получить 100% голосов. Обратите внимание, что каждый депутат в итоге получит вознаграждение (A_i , B_i или C_i).

Входные данные

В первой строке дано число N ($2 \leq N \leq 10^5$) — количество депутатов.

В следующей строке дано $N-1$ число p_i ($1 \leq p_i \leq N$), где p_i представляет собой номер прямого предка депутата $i+1$. Обратите внимание, что для первого депутата нет предка — он является вершиной дерева.

В следующих N строках дано по три целых числа A_i, B_i, C_i ($1 \leq A_i, B_i, C_i \leq 10^3$) — оплата, которую требует депутат i за голос, выступление и голосование под влиянием. Для каждого $1 \leq i \leq N$ выполняется $C_i \leq A_i \leq B_i$.

Выходные данные

Выведите одно число — минимально возможное количество денег, которые Василию нужно заплатить, чтобы получить 100% голосов.



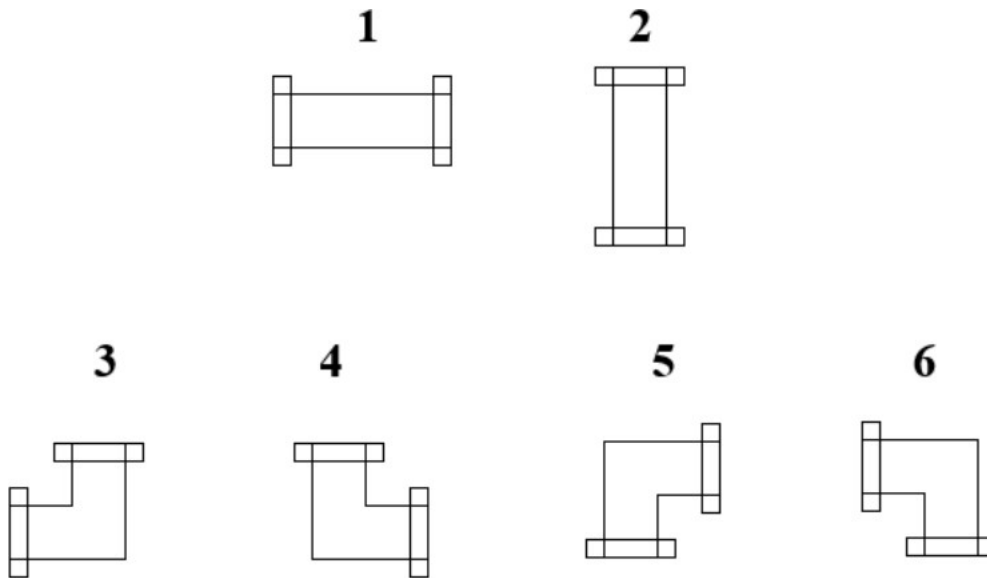
Пример

| Входные данные | Результат |
|---|-----------|
| 7 5 7 5 1 4 1 2 8 1 1 1 1 1 1 1 3 8 1 7 9 5 7 9 6 3 7 1 | 23 |

С. Труби

Обмеження на час роботи: для С та С++ — 1 секунда, для Java та Kotlin — 7 секунд
Обмеження на використання пам'яті: 256МБ

Система водопровідних труб в Ужляндії являє собою прямокутне поле розмірами $n \times m$, поділене на фрагменти. Фрагменти нумеруються зверху вниз від 1 до n та зліва направо від 1 до m . В кожному фрагменті може знаходитися труба одного із 6 типів (типи вказані на рисунку нижче). Якщо фрагмент порожній, то він позначається цифрою 0.



Водопровід вважається замкненим, якщо для кожної труби її кінець з'єднаний з кінцем якоїсь іншої труби.

Нещодавно вас призначили начальником водопроводу Ужляндії, тому у вас є повноваження повернути будь-яку трубу навколо своєї осі на кут, кратний 90 градусам. Ваше завдання — зробити цей водопровід замкненим, або повідомити, що це неможливо.

Вхідні данні

Перший рядок містить два числа: n та m ($1 \leq n, m \leq 1000$) — висоту та ширину поля.

Кожний з наступних n рядків містить по m цілих чисел $a_{i,1}, a_{i,2}, \dots, a_{i,m}$ ($0 \leq a_{i,j} \leq 6$). Кожне число описує тип труби, розташованої в клітинці з координатами (i, j) .

Вихідні данні

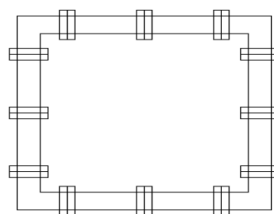
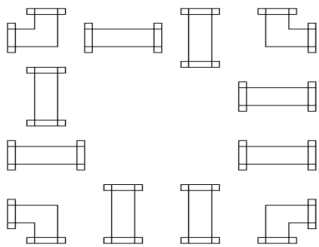
Виведіть «NO», якщо труби неможливо повернути так, щоб утворити замкнену систему, у протилежному випадку виведіть «YES», а далі виведіть n рядків по m чисел у кожному — опис замкненої системи, утвореної поворотами, у форматі, описаному вище.



Приклад

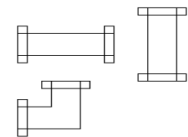
| Вхідні данні | Результат |
|--|--|
| <pre> 5 5 0 0 0 0 0 0 3 1 2 4 0 2 0 0 1 0 1 0 0 1 0 6 2 2 5 </pre> | <pre> YES 0 0 0 0 0 0 5 1 1 6 0 2 0 0 2 0 2 0 0 2 0 4 1 1 3 </pre> |
| <pre> 3 2 0 0 1 2 3 0 </pre> | <pre> NO </pre> |
| <pre> 8 8 0 4 2 1 2 2 3 0 0 2 5 4 0 0 1 0 0 1 4 5 0 0 1 0 0 1 0 0 3 2 5 0 0 2 5 3 4 2 1 5 0 1 2 5 5 0 0 1 0 2 5 2 5 0 0 1 0 4 2 1 2 2 2 5 </pre> | <pre> YES 0 5 1 1 1 1 6 0 0 2 5 6 0 0 2 0 0 2 4 3 0 0 2 0 0 2 0 0 5 1 3 0 0 2 5 6 4 1 1 6 0 2 2 4 6 0 0 2 0 2 4 1 3 0 0 2 0 4 1 1 1 1 1 3 </pre> |

Ілюстрація на рисунку нижче: початковий та замкнений (якщо він існує) стан.

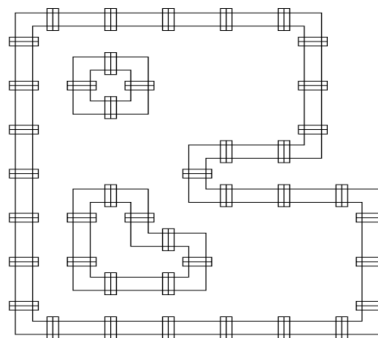
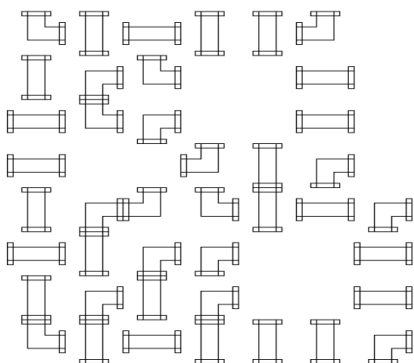


1

2



3





D. Lift

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Юрий решил сменить обстановку в своей квартире, которая находится на K -ом этаже L -этажного дома, оборудованного лифтом и лестницей. Для этого он закупил новую мебель, состоящую из T предметов.

Поскольку мебель тяжела для самостоятельного подъема в квартиру, он решил пригласить бригаду грузчиков. Чтобы поднять один из предметов мебели по лестнице на один этаж, требуется заплатить 300 грн., за спуск на один этаж — 150 грн. За подъем и спуск на лифте плата не взимается. Необходимость использования лестниц для транспортировки мебели обусловлена тем, что лифт может останавливаться только на каждом M -м этаже после первого (то есть на этажах с номерами $1, M+1, 2M+1, 3M+1, \dots$). За вызов грузчиков также придется дополнительно заплатить сумму, равную N грн. В эту сумму также входит разгрузка всей мебели из доставившей ее машины, перенос ее в дом на первый этаж, а также от лестницы или лифта в квартиру.

Требуется вычислить, какой минимальной суммы денег достаточно, чтобы грузчики доставили всю мебель в квартиру.

Входные данные

В единственной строке дано пять целых чисел, разделенных пробелами: N ($1 \leq N \leq 500$), T ($1 \leq T \leq 200$), L ($2 \leq L \leq 100$), K ($1 \leq K \leq L$) и M ($1 \leq M \leq L-1$).

Выходные данные

Выведите одно число — минимальную стоимость подъема всей мебели.

Пример

| Входные данные | Результат |
|----------------|-----------|
| 350 10 78 51 7 | 3350 |



Е. Число атомов

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Посчитайте число атомов в молекуле, заданной формулой вида C_2H_5OH .
Формула удовлетворяет регулярному выражению $([A-Z][a-z]? \setminus d\{0,3\})^+$.

Входные данные

В единственной строке входа находится непустая строка не длиннее 1000 символов, состоящая из цифр, больших и маленьких латинских букв.

Выходные данные

Выведите единственное целое число, ответ.

Пример

| Входные данные | Результат |
|------------------|-----------|
| He | 1 |
| H2O | 3 |
| C2H5OH | 9 |
| H00CC00H | 8 |
| C164H256Na2068S2 | 492 |

Пример 5 это майтотоксин. Он не поможет вам в решении задачи.



F. Kindergarten

Ограничение на время работы: для C и C++ — 1 секунда, для Java и Kotlin — 3 секунды
Ограничение на использование памяти: 256МБ

В детском саду есть $Nset$ шкафов для одежды детей, у каждого ребенка свой.

Каждый шкаф обозначен уникальным набором картинок: это объекты из нескольких разных категорий (например, персонажи сказок, посуда, цветные флажки и т.п.), не обязательно из каждой, но обязательно из разных, то есть из одной категории не более одной картинке. Наборы не могут отличаться только порядком картинок, а одна и та же картинка может использоваться в нескольких наборах.

Задано число категорий $Ncat$, для каждой категории дано число разных объектов (картинок) в ней $Nobj_i$.

Известно число уже существующих наборов картинок $Nset$ и состав каждого из них.

Когда в сад приходит новый ребенок, для него нужно составить новый набор с наименьшим возможным числом картинок (чтобы легче было находить свой шкафчик). Найдите это число, либо определите, что из данных объектов составить новый набор невозможно.

Входные данные

В первой строке дано число категорий $Ncat$ ($1 \leq Ncat \leq 4$).

Во второй строке даны $Ncat$ чисел $Nobj_i$ ($1 \leq Nobj_i \leq 15$) через пробел — количество объектов в каждой категории в порядке возрастания её номера.

В третьей строке дано число $Nset$ ($1 \leq Nset < 2^{16}$) — количество существующих наборов картинок.

Далее идёт $Nset$ строк, описывающих существующие наборы картинок. Каждая строка начинается с числа $Npics$ ($1 \leq Npics \leq Ncat$) — количества картинок в данном наборе. За ним следует через пробел $Npics$ слов-токенов, где маленькой латинской буквой (a, b ...) обозначен номер категории, а числом — номер объекта в ней. Например: «a4» — четвёртая картинка из первой категории, а «b11» — картинка 11 из второй категории. Картинки в каждом наборе даны в порядке возрастания номера категории.

Выходные данные

Выведите одно число — наименьшее возможное число картинок в новом наборе. Если из данных объектов составить новый набор невозможно, выведите «-1».



Пример

| Входные данные | Результат |
|--|-----------|
| 2 2 3 7 1 a1 2 a1 b2 2 a2 b1 1 b3 1 b2 1 a2 1 b1 | 2 |
| 3 2 3 1 7 1 a1 2 a1 b2 2 a2 b1 1 b1 1 b3 1 a2 2 b1 c1 | 1 |
| 2 2 1 5 2 a1 b1 2 a2 b1 1 b1 1 a2 1 a1 | -1 |

В первом примере все наборы из одной категории использованы, а наборы из двух категорий использованы не все (например, нет набора $\{a1, b1\}$), поэтому можно составить новый набор с двумя картинками.

Во втором примере ответ 1, так как использованы не все наборы из одной картинки.

В третьем примере число категорий 2; в первой из них 2 картинки, во второй 1, и все возможные наборы использованы.

Г. Три белых коня

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

В решающей партии матча американского гроссмейстера Бет Хармон и мальчика из чемодана Электроника Громова игра зашла в эндшпиль. Даже с проходной пешкой Хармон едва ли имела шансы на победу — компьютеры слишком хорошо считают в эндшпиле. Но американская шахматистка не растерялась и поставила перед советским игроком неожиданную задачу: проведя пешку, вместо ферзя, она на восьмой горизонтали поставила белого коня, третьего на доске.

— Этого нет в его оценочной функции! — ахнул профессор Громов, создатель Электроника.

Вам придется ее написать.

Какое количество разных свободных (от белых коней) полей на шахматной доске 8×8 бьют 3 белых коня с заданными разными координатами?

Входные данные

В единственной строке находятся позиции коней — 3 поля, разделенные одним пробелом. Поле состоит из 2 символов: первый — латинская буква от «a» до «h», второй — цифра от 1 до 8.

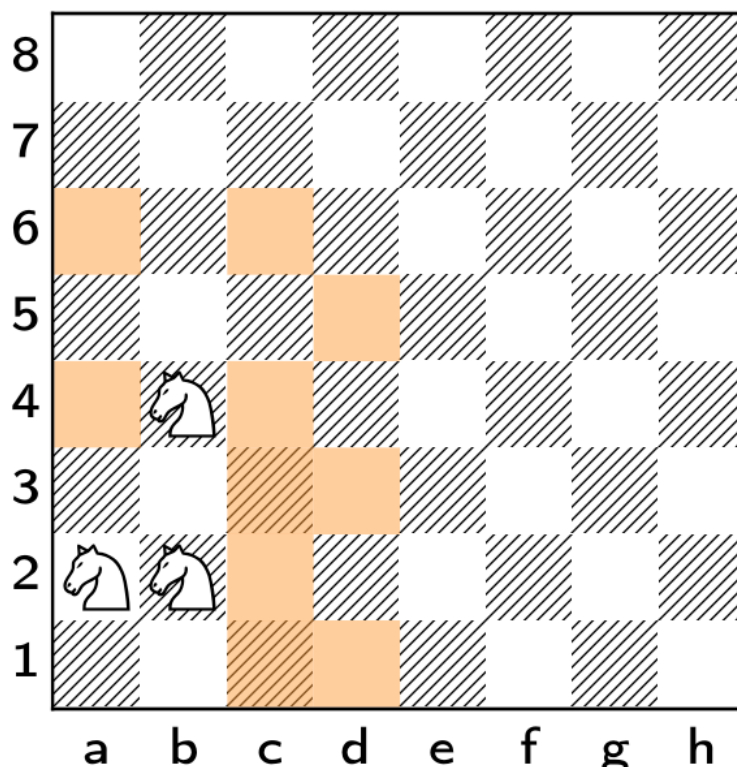
Выходные данные

Выведите единственное целое число, ответ.

Пример

| Входные данные | Результат |
|----------------|-----------|
| a2 b2 b4 | 10 |
| a1 a8 h8 | 6 |

В первом примере доска и считаемые поля выглядят так:





Н. Обхід міст

Обмеження на час роботи: для C та C++ — 1 секунда, для Java та Kotlin — 3 секунди
Обмеження на використання пам'яті: 256МБ

В Ужляндії є n міст і m двосторонніх доріг, що з'єднують між собою міста. Сергій живе в Ужляндії, і він дуже любить свою мальовничу країну, тому він вирішив поїхати у подорож і побувати в кожному із n міст. Щоб мандрівка не була нудною, він хоче побувати у кожному місті не більше одного разу. Починати і закінчувати маршрут він може в будь-якому із міст. Перед тим, як виїхати в подорож, він хоче дізнатися, скільки є можливих маршрутів, які його можуть зацікавити? Оскільки відповідь на задачу може бути дуже великою, виведіть її залишок від ділення на 2.

Вхідні данні

В першому рядку вхідного файлу містяться два цілих числа n і m ($1 \leq n \leq 1000$, $0 \leq m \leq n \cdot (n-1)/2$). В кожному із наступних m рядків містяться по 2 числа u і v ($1 \leq u, v \leq n$, $u \neq v$) — номери вершин, які з'єднує чергова дорога. Гарантується, що між кожною парою міст є не більше ніж 1 дорога, і що жодна дорога не з'єднує місто із самим собою.

Вихідні данні

Виведіть єдине число — відповідь на задачу.

Приклад

| Вхідні данні | Результат |
|--------------------------|-----------|
| 3 3 1 2 2 3 1 3 | 0 |

Існує 6 шляхів, які відповідають умові:

- 1 - 2 - 3
- 1 - 3 - 2
- 2 - 1 - 3
- 2 - 3 - 1
- 3 - 1 - 2
- 3 - 2 - 1

Отже, якщо кількість шляхів 6, то відповідь дорівнює $6 \bmod 2 = 0$.



I. Манхэттенский склад

Ограничение на время работы: для C и C++ — 1 секунда, для Java и Kotlin — 3 секунды
Ограничение на использование памяти: 256МБ

Как то раз служба доставки молока решила обосноваться на острове Манхэттен. У них уже были там клиенты, однако все еще не было своего склада. И с этим нужно срочно что-то делать, а то молоко прокисает.

Всего на острове есть N клиентов. Их координаты указаны в виде пар целых чисел x_i и y_i . Расположение всех клиентов различно, они расположены в углах квадратной сетки кварталов. Именно поэтому маршрут от склада до клиента — это всегда ломаная линия, состоящая из горизонтальных и вертикальных отрезков. Не удивительно, что в таких условиях расстояние вычисляется как сумма модулей разности координат склада и точки доставки.

Оптимально склад должен быть расположен так, чтоб сумма расстояний от склада до всех клиентов была как можно меньше:

$$\min_{x_s, y_s} \sum_{i=1}^N (|x_s - x_i| + |y_s - y_i|)$$

где x_s, y_s — координаты склада.

Найдите эту сумму. Из-за многоэтажной застройки можно считать, что координаты склада могут совпадать с каким-либо из клиентов.

Входные данные

В первой строке дано число клиентов N ($1 \leq N \leq 50\,000$).

Далее в N строках указаны пары целых чисел x_i, y_i ($-10\,000 \leq x_i, y_i \leq 10\,000$) — координаты i -го клиента.

Выходные данные

Вывести сумму расстояний до склада при условии его оптимального расположения. Ответ нужно округлять до целого по обычным математическим правилам.

Пример

| Входные данные | Результат |
|-------------------------------|-----------|
| 4 0 0 1 0 0 1 1 1 | 4 |



Ж. Перепутанные биты

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Папа отправил маленькой Элли Эрроузй сообщение из $N \leq 13$ маленьких латинских символов в ASCII. Из-за несовершенной конструкции радиоприемника, который собрала Элли, все биты перемешались и пришли A нулей и B единиц. Чтобы Элли перестала плакать, папа сообщил ей, что из всех возможных корректных сообщений с таким числом нулей и единиц отправленное им лексикографически K -ое (считая с 1). Определите это сообщение.

Входные данные

В единственной строке входа находятся 3 целых положительных числа: A , B и K ($A \leq 13 \cdot 8$, $B \leq 13 \cdot 8$, $K \leq 26^{13}$). Гарантируется, что A нулей и B единиц можно сгруппировать в хотя бы K разных корректных сообщений.

Выходные данные

В единственной строке выведите исходное сообщение.

Пример

| Входные данные | Результат |
|--------------------------|---------------|
| 5 3 4 | h |
| 35 37 405535840575 | pensacola |
| 50 54 192905298734071833 | smartestellie |

Корректных сообщений с 5 нулями и 3 единицами всего 5: «a», «b», «d», «h» и «p».

К. Обеспокоенные Онагатори

Ограничение на время работы: для C и C++ — 1 секунда, для Java и Kotlin — 4 секунды
Ограничение на использование памяти: 256МБ

Как известно, Онагатори — порода кур, которые обитают в Японии и обладают невероятно длинными хвостами. Где-то на побережье встали в ряд на расстоянии метра друг от друга N обеспокоенных Онагатори, которые убеждены, что совсем скоро обрушится цунами. Они уверены, что если каждый возьмется за чей-либо хвост, то это придаст им плавучесть, и они с легкостью перенесут любое цунами.



Каждый Онагатори может расправить свой хвост строго либо влево, либо вправо. Длина хвоста i -го Онагатори равна a_i метров. Если

i -й Онагатори расправит свой хвост влево, то за него сможет взяться любой Онагатори, который находится на позиции от $i - a_i$ до $i - 1$. Соответственно, если i -й Онагатори расправит хвост вправо, то от $i + 1$ до $i + a_i$.

Найдите такие направления хвостов, при которых каждый Онагатори будет держаться за чей-либо хвост, или скажите, что это невозможно. Порядок Онагатори не может меняться, потому что они напуганы и стоят на месте. За один хвост может взяться любое количество Онагатори (как ни один, так и все, кто дотягивается до хвоста). За свой хвост взяться нельзя. Обратите внимание, что каждый должен взяться за чей-либо хвост даже в том случае, если за их хвост кто-то взялся.

Входные данные

В первой строке дано число N ($2 \leq N \leq 300\,000$) — количество Онагатори. Во второй строке дан массив a из N целых чисел ($1 \leq a_i \leq N$), разделённых пробелами — длины хвостов.

Выходные данные

В первой строке выведите «YES» без кавычек, если существуют такие направления хвостов, при которых каждый сможет взяться за чей-либо хвост, или «NO» без кавычек, если ответа нет. Если ответ есть, то далее выведите строку длины N , состоящую только из символов «L» и «R», где «L» означает, что i -й хвост должен быть направлен влево, а «R» — вправо. Если ответов несколько — выведите любой.

Пример

| Входные данные | Результат |
|----------------|-------------|
| 5 1 2 3 1 5 | YES RLRL |
| 3 1 3 1 | NO |



L. Watchdog

Ограничение на время работы: для C и C++ — 1 секунда, для Java и Kotlin — 8 секунд
Ограничение на использование памяти: 256МБ

При постройке ранчо хозяин решил огородить пастбище для выпаса овец и поручить охрану собаке. Пастбище в плане представляет из себя многоугольник, а забор состоит только из нескольких отдельных жердей, между которыми не могут пройти овцы (наверное, на лучший забор хозяин поспешил). Поскольку такой забор может быть легко преодолен собакой, а соседи очень не любят когда на их территорию заходят чужие животные, хозяин задумал привязать собаку на цепь, закрепив ее внутри участка. Данная цепь будет препятствовать выходу собаки за пределы забора.

Хозяину требуется провести расчет длины цепи таким образом, чтобы при заданной точке привязки цепи собака имела доступ ко всей огороженной территории пастбища. Помогите фермеру, составив программу вычисления минимальной требуемой длины цепи. При перемещении собаки цепь может загибаться за углы забора, но ни собака, ни цепь не могут выходить за его пределы.

Входные данные

В первой строке дано одно число — количество углов в заборе N ($3 \leq N \leq 500$). Затем следует N строк, описывающих углы забора. Каждая строка содержит по два целых числа X_i и Y_i ($0 \leq X_i, Y_i \leq 10^6$) — координаты углов забора. Контур забора внутри пастбища не содержит самопересечений.

Затем в последней строке дано два целых числа: X_d и Y_d ($0 \leq X_d, Y_d \leq 10^6$) — координаты точки крепления конца цепи (данная точка находится внутри забора).

Выходные данные

Выведите одно число с точностью до двух знаков после запятой — минимальную длину цепи, которой с указанной точкой привязки хватит для доступа собаки в любую точку внутри пастбища.

Пример

| Входные данные | Результат |
|---|-----------|
| 5 1 1 2 4 6 5 7 2 4 3 2 2 | 5.40 |